-- select \*

-- from skyeng\_db.classes

-- select \*

-- from skyeng\_db.orders

-- select \*

-- from skyeng\_db.payments

-- select \*

-- from skyeng\_db.students

-- select \*

-- from skyeng\_db.teachers

--1.

with first\_payments

as (

select user\_id

, min(transaction\_datetime)::date as first\_payment\_date

from skyeng\_db.payments

where status\_name = 'success'

and operation\_name = 'Покупка уроков'

group by user\_id

)

-- 2.

, all\_dates

as (

select (generate\_series('2016-01-01', '2016-12-31', '1 day'::interval))::date as dt

)

-- 3.

, all\_dates\_by\_user

as (

select user\_id

, dt

from first\_payments fp

inner join all\_dates ad

on ad.dt >= fp.first\_payment\_date

)

-- 4.

, payments\_by\_dates

as (

select user\_id

, transaction\_datetime::date as payment\_date

, sum(classes) as transaction\_balance\_change

from skyeng\_db.payments

where status\_name = 'success'

and date\_part('year', transaction\_datetime) = 2016

group by user\_id

, payment\_date

)

-- 5.

, payments\_by\_dates\_cumsum

as (

select adu.user\_id

, dt

, transaction\_balance\_change

, sum(coalesce(transaction\_balance\_change, 0)) over (partition by adu.user\_id order by dt) as transaction\_balance\_change\_cs

from all\_dates\_by\_user adu

left join payments\_by\_dates pd

on adu.user\_id = pd.user\_id

and adu.dt = pd.payment\_date

)

-- 6.

, classes\_by\_dates

as (

select user\_id

, class\_start\_datetime::date as class\_date

, -1 \* count(\*) as classes\_balance\_change

from skyeng\_db.classes

where class\_status in ('success', 'failed\_by\_student')

and class\_type != 'trial'

and date\_part('year', class\_start\_datetime) = 2016

group by user\_id

, class\_date

)

-- 7.

, classes\_by\_dates\_dates\_cumsum

as (

select adu.user\_id

, dt

, classes\_balance\_change

, sum(coalesce(classes\_balance\_change, 0)) over (partition by adu.user\_id order by dt) as classes\_balance\_change\_cs

from all\_dates\_by\_user adu

left join classes\_by\_dates cd

on adu.user\_id = cd.user\_id

and adu.dt = cd.class\_date

)

-- 8.

, balances

as (

select pdc.dt

, transaction\_balance\_change

, transaction\_balance\_change\_cs

, classes\_balance\_change

, classes\_balance\_change\_cs

, transaction\_balance\_change\_cs + classes\_balance\_change\_cs as balance

from payments\_by\_dates\_cumsum pdc

inner join classes\_by\_dates\_dates\_cumsum cdc

on pdc.user\_id = cdc.user\_id

and pdc.dt = cdc.dt

-- where transaction\_balance\_change is not null

-- or classes\_balance\_change is not null

)

-- Задание 1. Выберите топ-1000 строк из CTE `balances`

-- select \*

-- from balances

-- order by user\_id

-- , dt

-- limit 1000

-- Студенты с успешными не пробными уроками, у которых нет успешных оплат, и их уроки

, users\_wo\_successful\_paymnets

as (

select distinct user\_id

from skyeng\_db.classes

where user\_id not in (

select distinct user\_id from skyeng\_db.payments

where date\_part('year', transaction\_datetime) = 2016 and status\_name = 'success'

)

and date\_part('year', class\_start\_datetime) = 2016 and class\_status in ('success', 'failed\_by\_student')

and class\_type != 'trial'

)

, classes\_wo\_successful\_paymnets

as (

select \*

from skyeng\_db.classes

where user\_id in (select \* from users\_wo\_successful\_paymnets)

and date\_part('year', class\_start\_datetime) = 2016

and class\_status in ('success', 'failed\_by\_student')

and class\_type != 'trial')

-- 9.

, final

as (

select dt

, sum(transaction\_balance\_change) as total\_transaction\_balance\_change

, sum(transaction\_balance\_change\_cs) as total\_transaction\_balance\_change\_cs

, sum(classes\_balance\_change) as total\_classes\_balance\_change

, sum(classes\_balance\_change\_cs) as total\_classes\_balance\_change\_cs

, sum(balance) as total\_balance

from balances

group by dt

order by dt

)

-- Задание 2.Создайте визуализацию

-- Сезонность уроков

, sum\_classes\_by\_dates

as (

select class\_date

, sum(classes\_balance\_change) as day\_classes\_balance\_change

from classes\_by\_dates

group by class\_date

)

, week\_day\_classes

as (

select extract(dow from class\_date) as week\_day

, min(day\_classes\_balance\_change) as max\_classes

, avg(day\_classes\_balance\_change) as avg\_classes

, max(day\_classes\_balance\_change) as min\_classes

from sum\_classes\_by\_dates

group by week\_day

order by week\_day

)

, week\_classes

as (

select date\_part('week', class\_date) as week

, min(day\_classes\_balance\_change) as max\_classes

, avg(day\_classes\_balance\_change) as avg\_classes

, max(day\_classes\_balance\_change) as min\_classes

from sum\_classes\_by\_dates

group by week

order by week

)

, month\_classes

as (

select date\_part('month', class\_date) as mnth

, min(day\_classes\_balance\_change) as max\_classes

, avg(day\_classes\_balance\_change) as avg\_classes

, max(day\_classes\_balance\_change) as min\_classes

from sum\_classes\_by\_dates

group by mnth

order by mnth

)

-- Сезонность оплат

, sum\_payments\_by\_dates

as (

select payment\_date

, sum(transaction\_balance\_change) as day\_transaction\_balance\_change

from payments\_by\_dates

group by payment\_date

)

, week\_day\_payments

as (

select extract(dow from payment\_date) as week\_day

, max(day\_transaction\_balance\_change) as max\_payments

, avg(day\_transaction\_balance\_change) as avg\_payments

, min(day\_transaction\_balance\_change) as min\_payments

from sum\_payments\_by\_dates

group by week\_day

order by week\_day

)

, week\_payments

as (

select date\_part('week', payment\_date) as week

, max(day\_transaction\_balance\_change) as max\_payments

, avg(day\_transaction\_balance\_change) as avg\_payments

, min(day\_transaction\_balance\_change) as min\_payments

from sum\_payments\_by\_dates

group by week

order by week

)

, month\_payments

as (

select date\_part('month', payment\_date) as mnth

, max(day\_transaction\_balance\_change) as max\_payments

, avg(day\_transaction\_balance\_change) as avg\_payments

, min(day\_transaction\_balance\_change) as min\_payments

from sum\_payments\_by\_dates

group by mnth

order by mnth

)

select \*

from final